

PCT/JP03/07476

12.06.03

Rec'd PCT/PTO

20 DEC 2004

日 本 国 特 許 庁

JAPAN PATENT OFFICE

REC'D 01 AUG 2003

WIPO PCT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日  
Date of Application: 2002年 7月 2日

出 願 番 号  
Application Number: 特願2002-193767  
[ST. 10/C]: [JP2002-193767]

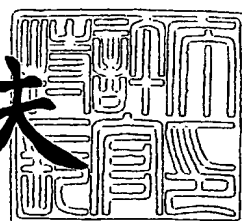
出 願 人  
Applicant(s): 三菱電機株式会社

PRIORITY DOCUMENT  
SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH  
RULE 17.1(a) OR (b)

2003年 7月18日

特許庁長官  
Commissioner,  
Japan Patent Office

今 井 康 夫



BEST AVAILABLE COPY

出証番号 出証特2003-3057155

【書類名】 特許願

【整理番号】 540737JP01

【提出日】 平成14年 7月 2日

【あて先】 特許庁長官殿

【国際特許分類】 H03M 13/00

【発明者】

【住所又は居所】 東京都千代田区丸の内二丁目2番3号 三菱電機株式会社  
社内

【氏名】 松本 渉

【特許出願人】

【識別番号】 000006013

【氏名又は名称】 三菱電機株式会社

【代理人】

【識別番号】 100089118

【弁理士】

【氏名又は名称】 酒井 宏明

【手数料の表示】

【予納台帳番号】 036711

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9803092

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 検査行列生成方法および検査行列生成装置

【特許請求の範囲】

【請求項1】 ユークリット幾何符号を用いて、列と行の重みまたはどちらか一方が均一でない低密度パリティ検査符号の検査行列を生成するための検査行列生成方法において、

符号長と符号化率を決定する符号長／符号化率決定ステップと、

ベースとなるユークリット幾何符号を選択するユークリット幾何符号選択ステップと、

選択したユークリット幾何符号を特定の関係式に基づいて並べ替えて基本行列を作成する並べ替えステップと、

条件「 $2 < \text{列の重みの最大値} \leq \text{ユークリット幾何符号における列内の1の数}$ 」を満たす列の重みの最大値を選択する最大重み選択ステップと、

行の重みの数を連続する2種類に限定した状態で、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な行の重みのセットを決定する第1の重み探索ステップと、

分割後の行数を考慮して、前記基本行列を底辺から順に削除する行削除ステップと、

前記行の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な列の重みのセットを決定する第2の重み探索ステップと、

前記行の重みのセットおよび前記列の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、低密度パリティ検査符号の行の重みと列の重みの最適なアンサンブルを探索する第3の重み探索ステップと、

最終的なアンサンブルに基づいて前記行削除後の基本行列の行および列の重みを所定の手順でランダムに分割する分割ステップと、

を含むことを特徴とする検査行列生成方法。

【請求項2】 前記並べ替えステップで用いる特定の関係式は、  
行列内の重みが列中の上部に配置されるように生成することを特徴とする請求  
項1に記載の検査行列生成方法。

【請求項3】 Cayleyグラフを用いて、列と行の重みまたはどちらか  
一方が均一でない低密度パリティ検査符号の検査行列を生成するための検査行列  
生成方法において、

符号長と符号化率を決定する符号長／符号化率決定ステップと、

ベースとなるCayleyグラフの行および列の重みを決定して基本行列を作  
成するCayleyグラフ決定ステップと、

条件「 $2 < \text{列の重みの最大値} \leq \text{Cayleyグラフにおける列内の1の数}$ 」を  
満たす列の重みの最大値を選択する最大重み選択ステップと、

行の重みの数を連続する2種類に限定した状態で、ガウス近似法による最適化  
を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサン  
ブルを探索し、その後、最適な行の重みのセットを決定する第1の重み探索ステ  
ップと、

分割後の行数を考慮して、前記基本行列を底辺から順に削除する行削除ステッ  
プと、

前記行の重みのセットを固定パラメータとして、ガウス近似法による最適化を  
用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサン  
ブルを探索し、その後、最適な列の重みのセットを決定する第2の重み探索ステッ  
プと、

前記行の重みのセットおよび前記列の重みのセットを固定パラメータとして、  
ガウス近似法による最適化を用いて、低密度パリティ検査符号の行の重みと列の  
重みの最適なアンサンブルを探索する第3の重み探索ステップと、

最終的なアンサンブルに基づいて前記行削除後の基本行列の行および列の重み  
を所定の手順でランダムに分割する分割ステップと、

を含むことを特徴とする検査行列生成方法。

【請求項4】 前記ガウス近似法では、符号化率を固定した状態で、かつガ  
ウスノイズが最大になるように、行の重みと列の重みの最適なアンサンブル (th

resholdが最小となるアンサンブル) を1回の線形計画法で探索することを特徴とする請求項1、2または3に記載の検査行列生成方法。

【請求項5】 前記第3の重み探索ステップでは、前記アンサンブルの重み配分を、重み単位の重み総数が整数で、かつ重み単位の重み総数の総和とユークリット幾何符号の「1」の総数とが等しくなるように調整し、

前記分割ステップでは、調整後のアンサンブルに基づいて分割処理を行うことを特徴とする請求項1～4のいずれか一つに記載の検査行列生成方法。

【請求項6】 前記分割ステップでは、基本のランダム系列のラテン方阵を作成し、当該ラテン方阵に基づいて、前記行削除後の基本行列における各行および各列から重み「1」を抽出することにより、各列および各行をランダムに分割することを特徴とする請求項1～5のいずれか一つに記載の検査行列生成方法。

【請求項7】 ユークリット幾何符号を用いて、列と行の重みまたはどちらか一方が均一でない低密度パリティ検査符号の検査行列を生成する検査行列生成装置において、

符号長と符号化率を決定する符号長／符号化率決定手段と、

ベースとなるユークリット幾何符号を選択するユークリット幾何符号選択手段と、

選択したユークリット幾何符号を特定の関係式に基づいて並べ替えて基本行列を作成する並べ替え手段と、

条件「 $2 < \text{列の重みの最大値} \leq \text{ユークリット幾何符号における列内の1の数}$ 」を満たす列の重みの最大値を選択する最大重み選択手段と、

行の重みの数を連続する2種類に限定した状態で、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な行の重みのセットを決定する第1の重み探索手段と、

分割後の行数を考慮して、前記基本行列を底辺から順に削除する行削除手段と、

前記行の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブル

ルを探索し、その後、最適な列の重みのセットを決定する第2の重み探索手段と

、

前記行の重みのセットおよび前記列の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、低密度パリティ検査符号の行の重みと列の重みの最適なアンサンブルを探索する第3の重み探索手段と、

最終的なアンサンブルに基づいて前記行削除後の基本行列の行および列の重みを所定の手順でランダムに分割する分割手段と、

を備えることを特徴とする検査行列生成装置。

【請求項8】 Cayleyグラフを用いて、列と行の重みまたはどちらか一方が均一でない低密度パリティ検査符号の検査行列を生成する検査行列生成装置において、

符号長と符号化率を決定する符号長／符号化率決定手段と、

ベースとなるCayleyグラフの行および列の重みを決定するCayleyグラフ決定手段と、

条件「 $2 < \text{列の重みの最大値} \leq \text{Cayleyグラフにおける列内の1の数}$ 」を満たす列の重みの最大値を選択する最大重み選択手段と、

行の重みの数を連続する2種類に限定した状態で、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な行の重みのセットを決定する第1の重み探索手段と、

分割後の行数を考慮して、前記基本行列を底辺から順に削除する行削除手段と

、

前記行の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な列の重みのセットを決定する第2の重み探索手段と

、

前記行の重みのセットおよび前記列の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、低密度パリティ検査符号の行の重みと列の重みの最適なアンサンブルを探索する第3の重み探索手段と、

最終的なアンサンブルに基づいて前記行削除後の基本行列の行および列の重みを所定の手順でランダムに分割する分割手段と、  
を備えることを特徴とする検査行列生成装置。

#### 【発明の詳細な説明】

##### 【0001】

#### 【発明の属する技術分野】

本発明は、誤り訂正符号として低密度パリティ検査（LDPC：Low-Density Parity-Check）符号を採用した場合における検査行列生成方法および検査行列生成装置に関するものであり、特に、確定的で特性が安定したLDPC符号用検査行列を探索可能な検査行列生成方法および検査行列生成装置に関するものである。

##### 【0002】

#### 【従来の技術】

以下、従来のLDPC符号用検査行列生成方法について説明する。図13は、LDPC符号化／復号システムを示す図である。図13において、101は符号化器であり、102は変調器であり、103は通信路であり、104は復調器であり、105は復号器である。ここでは、従来のLDPC符号用検査行列生成方法を説明する前に、LDPC符号を使用した場合の符号化、復号の流れについて説明する。

##### 【0003】

まず、送信側の符号化器101では、後述する所定の方法で検査行列Hを生成する。そして、以下の条件に基づいて生成行列Gを求める。

$G: k \times n$  行列 ( $k$ : 情報長,  $n$ : 符号語長)

$GH^T = 0$  ( $T$ は転置行列)

##### 【0004】

その後、符号化器101では、情報長 $k$ のメッセージ( $m_1 m_2 \dots m_k$ )を受け取り、上記生成行列Gを用いて符号語Cを生成する。

$C = (m_1 m_2 \dots m_k) G$

$= (c_1 c_2 \dots c_n)$  (ただし、 $H(c_1 c_2 \dots c_n)^T = 0$ )

## 【0005】

そして、変調器102では、生成した符号語Cに対して、BPSK, QPSK, 多値QAMなどのデジタル変調を行い、送信する。

## 【0006】

一方、受信側では、復調器104が、通信路103を介して受け取った変調信号に対して、BPSK, QPSK, 多値QAMなどのデジタル復調を行い、さらに、復号器105が、LDPC符号化された復調結果に対して、「sum-productアルゴリズム」による繰り返し復号を実施し、推定結果（もとの $m_1$   $m_2 \dots m_k$ に対応）を出力する。

## 【0007】

ここで、従来のLDPC符号用検査行列生成方法について具体的に説明する。LDPC符号用の検査行列としては、たとえば、LDPCの発案者Gallagerにより以下のような行列が提案されている（図14参照）。

## 【0008】

図14に示す行列は、「1」と「0」の2値の行列で、「1」の部分塗りつぶしている。他の部分は全て「0」である。この行列は、1行の「1」の数（これを行の重みと表現する）が4で、1列の「1」の数（これを列の重みと表現する）が3であり、全ての列と行の重みが均一なため、これを一般に「Regular-LDPC符号」と呼んでいる。また、Gallagerの符号では、たとえば、図14に示すように、行列を3ブロックに分け、2ブロック目と3ブロック目に対してランダム置換を行っている。

## 【0009】

しかしながら、このランダム置換には、所定のルールがないため、より特性の良好な符号を見つけるためには、計算機による時間のかかる探索を行わなければならない。

## 【0010】

そこで、たとえば、計算機探索によらなくても確定的に行列を生成でき、比較的安定した良好な特性を示すLDPC符号として、ユークリッド幾何符号を用いる方法が、Y. Kou等（Y. Kou, S. Lin, and M. P. C. Fossorier, "Low Den



sity Parity Check Codes Based on Finite Geometries: A Rediscovery," ISIT 2000, pp. 200, Sorrento, Italy, June 25-30, 2000.) によって提案された。この方法では、規則的な *ensemble* (アンサンブル) で構成された「Regular-LDPC符号」について説明されている。

#### 【0011】

ここでは、有限幾何符号の一種であるユークリット幾何符号  $EG(2, 2^6)$  を用いて LDPC 符号の検査行列を生成する方法が提案されており、誤り率  $10^{-4}$  点において、シャノン限界から  $1.45 \text{ dB}$  に接近した特性を得ている。図 15 は、たとえば、ユークリット幾何符号  $EG(2, 2^2)$  の構成を示す図であり、行、列のそれぞれの重みが 4, 4 の「Regular-LDPC符号」構造をしている。

#### 【0012】

したがって、ユークリット幾何符号  $EG(m, 2^s)$  の場合、その特性は、以下のように規定される。

符号長:  $n = 2^{2s} - 1$   
冗長ビット長:  $n - k = 3^s - 1$   
情報長:  $k = 2^{2s} - 3^s$   
最小距離:  $d_{\min} = 2^s + 1$   
密度:  $r = 2^s / (2^{2s} - 1)$

#### 【0013】

図 15 を見ても分かるように、ユークリット幾何符号は、各行の「1」の配置が行毎に巡回シフトした構造になっており、符号が容易にかつ確定的に構成できる特長がある。

#### 【0014】

Y. Kou らによる検査行列の生成方法では、さらに、上記ユークリット幾何符号に基づいて行と列の重みを変更し、行、列を必要に応じて拡張している。たとえば、 $EG(2, 2^2)$  の列の重みを  $1/2$  に分離する場合、Y. Kou らの論文では、1列内に4つある重みを1つ置きに2個ずつ分離する。図 16 は、列の重みを 4 から 2 に規則的に分離した例を示す図である。

## 【0015】

一方、上記「Regular-LDPC符号」の特性よりも「Irregular-LDPC符号」の特性の方が良好であることが、Luby等 (M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Improved Low-Density Parity-Check Codes Using Irregular Graphs and Belief Propagation," Proceedings of 1998 IEEE International Symposium on Information Theory, pp. 171, Cambridge, Mass., August 16-21, 1998.) により報告された。なお、上記「Irregular-LDPC符号」は、列と行の重みがそれぞれあるいはどちらか一方が均一でないLDPC符号を表す。

## 【0016】

そして、それは、Richardson等 (T. J. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," IEEE Trans. Inform. Theory, vol.47, No.2, pp.599-618, Feb. 2001.)、あるいはChung等 (S.-Y. Chung, T. J. Richardson, and R. Urbanke, "Analysis of Sum-Product Decoding of Low-Density Parity-Check Codes Using a Gaussian Approximation," IEEE Trans. Inform. Theory, vol.47, No.2, pp.657-670, Feb. 2001.) によって理論的に解析された。

## 【0017】

特に、Chung等は、繰り返し復号器における入力と出力の対数尤度比 (LLR) がガウス分布に近似できると仮定してLDPC符号の「Sum-Product アルゴリズム」を解析し、良好な行と列の重みのアンサンブルを求めている。

## 【0018】

## 【発明が解決しようとする課題】

しかしながら、たとえば、上記Chung等による従来のLDPC符号用検査行列生成方法は、行内の「1」の点の数（後述するバリエابلノードの次数配分に相当）と、列内の「1」の点の数（後述するチェックノードの次数配分に相当）と、の両方を変数として、下記の(1)式 (rate: 符号化率) が最大となるバリエابلノードの次数配分およびチェックノードの次数配分を求めている。

すなわち、SNR (Signal to Noise Ratio) が最小となるアンサンブルを線形計画法により探索している。

【0019】

【数1】

$$\text{rate} = 1 - \frac{\int_0^1 \rho(x) dx}{\int_0^1 \lambda(x) dx} \quad \dots(1)$$

【0020】

そのため、上記「rate」の最大値により得られる検査行列が流動的になり、特性が安定しない、という問題があった。また、従来のLDPC符号用検査行列生成方法は、バリエブルノードの次数配分の導出とチェックノードの次数配分の導出とを所定回数にわたって繰り返し行っているため、探索処理にある程度の時間を要する、という問題もあった。

【0021】

本発明は、上記に鑑みてなされたものであって、確定的で特性が安定し、かつ任意のアンサンブルに対応したLDPC符号用の検査行列を容易に探索可能で、さらに、性能の良好な検査行列生成方法および検査行列生成装置を得ることを目的とする。

【0022】

【課題を解決するための手段】

上述した課題を解決し、目的を達成するために、本発明にかかる検査行列生成方法にあつては、ユークリット幾何符号を用いて、列と行の重みまたはどちらか一方が均一でない低密度パリティ検査符号の検査行列を生成するために、符号長と符号化率を決定する符号長／符号化率決定ステップと、ベースとなるユークリット幾何符号を選択するユークリット幾何符号選択ステップと、選択したユークリット幾何符号を特定の関係式に基づいて並べ替えて基本行列を作成する並べ替えステップと、条件「 $2 < \text{列の重みの最大値} \leq \text{ユークリット幾何符号における列内の1の数}$ 」を満たす列の重みの最大値を選択する最大重み選択ステップと、行の重みの数を連続する2種類に限定した状態で、ガウス近似法による最適化を用

いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な行の重みのセットを決定する第1の重み探索ステップと、分割後の行数を考慮して、前記基本行列を底辺から順に削除する行削除ステップと、前記行の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な列の重みのセットを決定する第2の重み探索ステップと、前記行の重みのセットおよび前記列の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、低密度パリティ検査符号の行の重みと列の重みの最適なアンサンブルを探索する第3の重み探索ステップと、最終的なアンサンブルに基づいて前記行削除後の基本行列の行および列の重みを所定の手順でランダムに分割する分割ステップと、を含むことを特徴とする。

#### 【0023】

つぎの発明にかかる検査行列生成方法において、前記並べ替えステップで用いる特定の関係式は、行列内の重みが列中の上部に配置されるように生成することを特徴とする。

#### 【0024】

つぎの発明にかかる検査行列生成方法にあつては、Cayleyグラフを用いて、列と行の重みまたはどちらか一方が均一でない低密度パリティ検査符号の検査行列を生成するために、符号長と符号化率を決定する符号長／符号化率決定ステップと、ベースとなるCayleyグラフの行および列の重みを決定して基本行列を作成するCayleyグラフ決定ステップと、条件「 $2 < \text{列の重みの最大値} \leq \text{Cayleyグラフにおける列内の1の数}$ 」を満たす列の重みの最大値を選択する最大重み選択ステップと、行の重みの数を連続する2種類に限定した状態で、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な行の重みのセットを決定する第1の重み探索ステップと、分割後の行数を考慮して、前記基本行列を底辺から順に削除する行削除ステップと、前記行の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な列の重み

のセットを決定する第2の重み探索ステップと、前記行の重みのセットおよび前記列の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、低密度パリティ検査符号の行の重みと列の重みの最適なアンサンブルを探索する第3の重み探索ステップと、最終的なアンサンブルに基づいて前記行削除後の基本行列の行および列の重みを所定の手順でランダムに分割する分割ステップと、を含むことを特徴とする。

#### 【0025】

つぎの発明にかかる検査行列生成方法において、前記ガウス近似法では、符号化率を固定した状態で、かつガウスノイズが最大になるように、行の重みと列の重みの最適なアンサンブル (thresholdが最小となるアンサンブル) を1回の線形計画法で探索することを特徴とする。

#### 【0026】

つぎの発明にかかる検査行列生成方法において、前記第3の重み探索ステップでは、前記アンサンブルの重み配分を、重み単位の重み総数が整数で、かつ重み単位の重み総数の総和とユークリット幾何符号の「1」の総数とが等しくなるように調整し、前記分割ステップでは、調整後のアンサンブルに基づいて分割処理を行うことを特徴とする。

#### 【0027】

つぎの発明にかかる検査行列生成方法において、前記分割ステップでは、基本のランダム系列のラテン方阵を作成し、当該ラテン方阵に基づいて、前記行削除後の基本行列における各行および各列から重み「1」を抽出することにより、各列および各行をランダムに分割することを特徴とする。

#### 【0028】

つぎの発明にかかる検査行列生成装置にあつては、ユークリット幾何符号を用いて、列と行の重みまたはどちらか一方が均一でない低密度パリティ検査符号の検査行列を生成する構成として、符号長と符号化率を決定する符号長／符号化率決定手段と、ベースとなるユークリット幾何符号を選択するユークリット幾何符号選択手段と、選択したユークリット幾何符号を特定の関係式に基づいて並べ替えて基本行列を作成する並べ替え手段と、条件「 $2 < \text{列の重みの最大値} \leq \text{ユーク}$

リット幾何符号における列内の 1 の数」を満たす列の重みの最大値を選択する最大重み選択手段と、行の重みの数を連続する 2 種類に限定した状態で、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な行の重みのセットを決定する第 1 の重み探索手段と、分割後の行数を考慮して、前記基本行列を底辺から順に削除する行削除手段と、前記行の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な列の重みのセットを決定する第 2 の重み探索手段と、前記行の重みのセットおよび前記列の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、低密度パリティ検査符号の行の重みと列の重みの最適なアンサンブルを探索する第 3 の重み探索手段と、最終的なアンサンブルに基づいて前記行削除後の基本行列の行および列の重みを所定の手順でランダムに分割する分割手段と、を備えることを特徴とする。

#### 【0029】

つぎの発明にかかる検査行列生成装置にあつては、Cayley グラフを用いて、列と行の重みまたはどちらか一方が均一でない低密度パリティ検査符号の検査行列を生成する構成として、符号長と符号化率を決定する符号長／符号化率決定手段と、ベースとなる Cayley グラフの行および列の重みを決定する Cayley グラフ決定手段と、条件「 $2 < \text{列の重みの最大値} \leq \text{Cayley グラフにおける列内の 1 の数}$ 」を満たす列の重みの最大値を選択する最大重み選択手段と、行の重みの数を連続する 2 種類に限定した状態で、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な行の重みのセットを決定する第 1 の重み探索手段と、分割後の行数を考慮して、前記基本行列を底辺から順に削除する行削除手段と、前記行の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、暫定的に、低密度パリティ検査符号の行の重みと列の重みのアンサンブルを探索し、その後、最適な列の重みのセットを決定する第 2 の重み探索手段と、前記行の重みのセットおよび前記列の重みのセットを固定パラメータとして、ガウス近似法による最適化を用いて、低密度パリティ検査符号の行の重みと

列の重みの最適なアンサンプルを探索する第3の重み探索手段と、最終的なアンサンプルに基づいて前記行削除後の基本行列の行および列の重みを所定の手順でランダムに分割する分割手段と、を備えることを特徴とする。

### 【0030】

#### 【発明の実施の形態】

以下に、本発明にかかる検査行列生成方法および検査行列生成装置の実施の形態を図面に基づいて詳細に説明する。なお、この実施の形態によりこの発明が限定されるものではない。

### 【0031】

図1は、本発明にかかるLDPC符号用検査行列生成方法を示すフローチャートである。なお、本実施の形態におけるLDPC符号用検査行列生成方法は、たとえば、設定されるパラメータに応じて通信装置内で実行する構成としてもよいし、通信装置外部の他の制御装置（計算機等）で実行することとしてもよい。本実施の形態におけるLDPC符号用検査行列生成方法が通信装置外部で実行される場合は、生成済みのLDPC符号用検査行列が通信装置に格納される。以降の実施の形態では、説明の便宜上、通信装置内で上記方法を実行する場合について説明する。

### 【0032】

まず、本実施の形態のLDPC符号用検査行列生成方法を説明する前に、本実施の形態のLDPC符号用検査行列生成方法を実現可能な符号化器および復号器の位置付け、および「Irregular-LDPC符号」用の従来の検査行列生成方法について説明する。なお、LDPC符号化／復号システムの構成については、先に説明した図13と同様である。

### 【0033】

送信側の符号化器101では、後述する本実施の形態のLDPC符号用検査行列生成方法で検査行列Hを生成する。そして、以下の条件に基づいて生成行列Gを求める。

$G: k \times n$  行列 ( $k$ : 情報長,  $n$ : 符号語長)

$GH^T = 0$  ( $T$ は転置行列)

## 【0034】

その後、符号化器101では、情報長 $k$ のメッセージ $(m_1 m_2 \dots m_k)$ を受け取り、上記生成行列 $G$ を用いて符号語 $C$ を生成する。

$$\begin{aligned} C &= (m_1 m_2 \dots m_k) G \\ &= (c_1 c_2 \dots c_n) \quad (\text{ただし、} H(c_1 c_2 \dots c_n)^T = 0) \end{aligned}$$

## 【0035】

そして、変調器102では、生成した符号語 $C$ に対して、BPSK, QPSK, 多値QAMなどのデジタル変調を行い、送信する。

## 【0036】

一方、受信側では、復調器104が、通信路103を介して受け取った変調信号に対して、BPSK, QPSK, 多値QAMなどのデジタル復調を行い、さらに、復号器105が、LDPC符号化された復調結果に対して、「sum-product アルゴリズム」による繰り返し復号を実施し、推定結果（もとの $m_1 m_2 \dots m_k$ に対応）を出力する。

## 【0037】

つぎに、Chung等 (S.-Y. Chung, T. J. Richardson, and R. Urbanke, "Analysis of Sum-Product Decoding of Low-Density Parity-Check Codes Using a Gaussian Approximation," IEEE Trans. Inform. Theory, vol.47, No.2, pp.657-670, Feb. 2001.) によって理論的に解析された、「Irregular LDPC符号」用の従来の検査行列生成方法について詳細に説明する。ここでは、繰り返し復号器における入力と出力の対数尤度比 (LLR) がガウス分布に近似できると仮定してLDPC符号の「Sum-Product アルゴリズム」を解析し、良好な行と列の重みのアンサンブルを求めている。

## 【0038】

なお、上記論文に記述されたLDPC符号用検査行列生成方法であるガウス近似法 (Gaussian Approximation) では、前提として、検査行列における行内の「1」の点をバリエブルノードと定義し、列内の「1」の点をチェックノードと定義する。

## 【0039】



まず、チェックノードからバリエブルノードへのLLRメッセージ伝搬を解析する。 $0 < s < \infty$ と $0 \leq t < \infty$ という条件において、以下の関数(2)式を定義する。なお、 $s = m_{u0}$ は $u_0$ の平均値であり、 $u_0$ は分散値 $\sigma_n^2$ のガウスノイズを含む伝送路を経由して受信した信号の対数尤度比(LLR)であり、 $t$ は所定の繰り返しの時点におけるチェックノードのLLR出力値のアンサンブル平均である。

【0040】

【数2】

$$f_j(s, t) = \phi^{-1} \left( 1 - \left[ 1 - \sum_{i=2}^{d_l} \lambda_i \phi(s + (i-1)t) \right]^{j-1} \right)$$

$$f(s, t) = \sum_{j=2}^{d_r} \rho_j f_j(s, t) \quad \dots(2)$$

【0041】

なお、上記 $\lambda(x)$ および $\rho(x)$ は、それぞれバリエブルノードおよびチェックノードの次数配分(バリエブルノードとチェックノードの各1行、各1列内の「1」の数を次数と表現する)の生成関数を表し、(3)式および(4)式のように表すことができる。また、 $\lambda_i$ と $\rho_i$ は、それぞれ次数 $i$ のバリエブルノードとチェックノードに属するエッジの比率を表す。また、 $d_l$ は最大バリエブルノードの次数であり、 $d_r$ は最大チェックノードの次数である。

【0042】

【数3】

$$\lambda(x) = \sum_{i=2}^{d_l} \lambda_i x^{i-1} \quad \dots(3)$$

【0043】

【数4】

$$\rho(x) = \sum_{i=2}^{d_r} \rho_i x^{i-1} \quad \dots(4)$$

【0044】

ただし、 $\phi(x)$  は下記 (5) 式のように定義する。

【0045】

【数5】

$$\phi(x) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi x}} \int_R \tanh \frac{u}{2} \cdot e^{-\frac{(u-x)^2}{4x}} du & \text{if } x > 0 \\ 1 & \text{if } x \leq 0 \end{cases} \quad \dots(5)$$

【0046】

そして、(2) 式は、等価的に下記 (6) 式と表すことができる。

【0047】

【数6】

$$t_1 = f(s, t_{1-1}) \quad \dots(6)$$

【0048】

なお、 $t_1$  は 1 番目の繰り返し時点におけるチェックノードの LLR 出力値のアンサンブル平均である。

【0049】

ここで、誤りが 0 となりうる SNR の限界 (threshold) を求めるための条件は、 $1 \rightarrow \infty$  のときに  $t_1(s) \rightarrow \infty$  ( $R^+$  と表現する) となることであり、この条件を満たすためには、以下の条件 (7) 式を満たす必要がある。

【0050】

【数7】

$$t < f(s, t), \text{ 全ての } t \in R^+ \quad \dots(7)$$

【0051】

つぎに、バリアブルノードからチェックノードへの LLR メッセージ伝搬を解析する。 $0 < s < \infty$  と  $0 < r \leq 1$  という条件において、以下の関数 (8) 式を定義する。なお、 $r$  の初期値  $r_0$  は  $\phi(s)$  である。

【0052】

【数8】

$$h_i(s, r) = \phi \left( s + (i-1) \sum_{j=2}^{d_r} \rho_j \phi(1 - (1-r)^{j-1}) \right)$$

$$h(s, r) = \sum_{i=2}^{d_l} \lambda_i h_i(s, r) \quad \dots(8)$$

【0053】

そして、(8)式は、等価的に下記(9)式と表すことができる。

【0054】

【数9】

$$r_1 = h(s, r_{l-1}) \quad \dots(9)$$

【0055】

ここで、誤りが0となりうるSNRの限界(threshold)を求めるための条件は、 $r_1(s) \rightarrow 0$ となることであり、この条件を満たすためには、以下の条件(10)式を満たす必要がある。

【0056】

【数10】

$$r > h(s, r), \text{ 全ての } r \in (0, \phi(s)) \quad \dots(10)$$

【0057】

さらに、上記Chung等の論文では、上記式を用いて以下の手順でバリアブルノードとチェックノードの最適な次数を探索している(ガウス近似法)。

(1) 生成関数 $\lambda(x)$ とガウスノイズ $\sigma_n$ が与えられていると仮定し、生成関数 $\rho(x)$ を変数として、前述した(1)式が最大となる点を探索する。なお、この探索における拘束条件は、 $\rho(1) = 1$ と正規化することと、上記(7)式を満たすことである。

(2) 生成関数 $\rho(x)$ とガウスノイズ $\sigma_n$ が与えられていると仮定し(たとえ

ば、(1)の結果より得られる値)、生成関数 $\lambda(x)$ を変数として、(1)式が最大となる点を探索する。なお、この探索における拘束条件は、 $\lambda(1) = 1$ と正規化することと、上記(10)式を満たすことである。

(3) 最大「rate」を求めるために、上記(1)と上記(2)を繰り返し実行し、生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ のより良好なアンサンブルを線形計画法で探索する。

(4) 最後に、ガウスノイズ $\sigma_n$ より信号電力を1と正規化して、SNRの限界(threshold)を求める((11)式参照)。

【0058】

【数11】

$$\text{threshold(dB)} = -10 * \log_{10}(2 * \sigma_n^2) \quad \dots(11)$$

【0059】

しかしながら、上記Chung等の論文では、「rate (符号化率)」の最大値により得られる検査行列が流動的になり、設計時の仕様として固定されるrateが変動してしまう、という問題があった。また、上記Chung等の論文では、バリエブルノードの次数配分の導出とチェックノードの次数配分の導出とを所定回数にわたって繰り返し行っているため、探索処理にある程度の時間を要する、という問題や、任意のアンサンブル、任意の符号長、任意の符号化率に容易に対応することができない、という問題もあった。

【0060】

そこで、本実施の形態においては、確定的で特性が安定し、かつ任意のアンサンブル、任意の符号長、任意の符号化率に対応した「Irregular-LDPC符号」用の検査行列を、短時間で容易に探索する方法について説明する(図1参照)。具体的にいうと、ここでは、ユークリット幾何符号における1行または1列の「1」の配置を分割および削除することにより、「Irregular-LDPC符号」用の検査行列を生成する。図1は、実施の形態1のLDPC符号用検査行列生成方法を示す図である。

【0061】

本実施の形態のLDPC符号用検査行列生成方法では、まず、符号長 $N$ と符号化率 $rate$ を決定する(ステップS1)。これにより、検査行列のサイズが $N \times M$ に確定する。なお、 $M$ は $N \times (1 - rate)$ で表すことができる。したがって、たとえば、 $N=6000$ 、 $rate=0.5$ の場合、 $M$ は $M=6000 \times 0.5=3000$ となる。

#### 【0062】

つぎに、「Irregular-LDPC符号」用の検査行列のベースとなるユークリット幾何符号 $EG(2, 2^s)$ を選択する(ステップS2)。ここでは、行の重みと列の重みがそれぞれ $2^s$ となる。

#### 【0063】

つぎに、選択したユークリット幾何符号 $EG(2, 2^s)$ を、列内の1の位置が列中のできるだけ上部にくるように、以下の手順で並べ替える(ステップS3)。まず、この並べ替え手順を一般的に表現すると、 $h_k(X) \in \text{ガロア体GF}(2^{2s})$ 、 $k = \{1, 2, \dots, 4 \times (2^{2s}-1)\}$ の場合、下記(12)式のよう表現できる。

#### 【0064】

#### 【数12】

$$\begin{bmatrix} h_{i+0}(X) \\ h_{i+1}(X) \\ h_{i+2}(X) \\ \vdots \\ h_{i+(2^s-1)}(X) \end{bmatrix} = \begin{bmatrix} 1 \\ X^{-w1} \\ X^{-w2} \\ \vdots \\ X^{-w(2^s-1)} \end{bmatrix} \times \left[ \left( 1 + X^{w1} + X^{w2} + \dots + X^{w(2^s-1)} \right) \cdot X^{(i-1)} \right] \dots (12)$$

#### 【0065】

なお、 $i=1 \sim 2^{2s}-1$ とする。また、(12)式の $(\cdot)$ 内の多項式は、ユークリット幾何符号 $EG(2, 2^s)$ の最初の行を表現した式を表す。

#### 【0066】

そして、 $i = 1 \sim 2^{2s} - 1$ ,  $j = 1 \sim i - 1$  までの間に、 $h_i(X) = h_j(X)$  が存在する場合は、 $h_i(X)$  を削除する。この並べ替え処理により、後述する行の削除処理（ステップ S 6）を行う場合に、できるだけ重みの大きい列を残すことができ、かつ列内の重みのバリエーションをできるだけ少なくすることができる。

#### 【0067】

具体例として、たとえば、 $s = 2$  とした場合、すなわち、ユークリット幾何符号 EG (2,  $2^2$ ) を選択した場合、上記並べ替え手順を実施すると、図 2 のマトリクスが図 3 のマトリクスのように並べ替えられる。図 2 は、ステップ S 2 にて選択されたユークリット幾何符号 EG (2,  $2^2$ ) のマトリクスを示す図（空白は 0 を表す）であり、図 3 は、並べ替え後のマトリクス（基本行列）を示す図である。

#### 【0068】

つぎに、列の最大重み  $\gamma_1$  ( $2 < \gamma_1 \leq 2^s$ ) を選択する（ステップ S 4）。LDPC 符号を用いた符号化／復号においては、一般的に、2 部グラフ上に「サイクル 4」および「サイクル 6」が少ないほど良好な特性を得ることができる。したがって、LDPC 符号としては、「サイクル 4」や「サイクル 6」といった少ないサイクルの発生を抑制する構造が望ましい。ユークリット幾何符号には、すでに「サイクル 4」が存在しないので、「サイクル 6」を削減することによって、復号特性の向上を図る。たとえば、ユークリット幾何符号 EG (2,  $2^5$ ) の場合には、重みが 12 程度のときに「サイクル 6」の数が減少し、よい特性を示すことが一般に知られているので、ここでは、 $\gamma_1 = 12$  を選択する。なお、ここでは、 $\gamma_1 = 12$  を選択することとしたが、これに限らず、上記条件を満たしていれば、これ以外の数を選択してもよい。

#### 【0069】

つぎに、ガウス近似法による最適化を用いて、暫定的に、要求された符号化率に基づく「Irregular-LDPC 符号」のアンサンブルを求める（ステップ S 5）。なお、行の重み配分の生成関数  $\rho(x)$  は  $\rho(x) = \rho_j x^{j-1} + (1 - \rho_j) x^j$  とする。ただし、 $j$  は  $j \geq 2$  の整数である。

## 【0070】

そして、下記の(13)式を満たすパラメータセット  $\{\mu_1, \mu_2\}$  を選択する(ステップS5)。なお、 $2 \leq \mu_i \leq 2^s$ ,  $i \in \{1, 2\}$  である。

## 【0071】

## 【数13】

$$\begin{aligned} b_1\mu_1 + b_2\mu_2 &= 2^s \\ \arg \cdot \min |\rho_{\mu_i} - \rho'_{\mu_i}| \\ \rho'_{\mu_i} &= \frac{\mu_i \times b_i}{2^s} \end{aligned} \quad \dots(13)$$

## 【0072】

ただし、 $b_i$ ,  $i$  は非負の整数であり、 $\mu_i$  は行の重みを表し、 $\rho_{\mu_i}'$  は「Irregular-LDPC符号」にあわせて調整した重み $\mu_i$ に属するエッジの比率を表す。ここでは、パラメータセットとして、 $\mu_1=8$ ,  $\mu_2=9$ を選択する。図4は、 $\gamma_1=d_1=12$ ,  $rate=0.5$ の場合の、重み配分後の生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ のアンサンブル(暫定)を示す図である。 $x$ は重みを表し、 $\lambda_x$ および $\rho_x$ はそれぞれバリアブルノードとチェックノードの重み配分を表す。また、表中 $\sigma_{GA}$ はガウス近似法により導出した「threshold」時のノイズ分散値を表し、 $SNR_{norm}(GA)$ はガウス近似法により導出した「threshold」時のSNRとシャノン限界のSNRとの差分を表す。

## 【0073】

ここで、バリアブルノードの次数配分の生成関数 $\lambda(x)$ とチェックノードの次数配分の生成関数 $\rho(x)$ のアンサンブルを探索するための本実施の形態のガウス近似法の実行手順について説明する。

- (1) 符号化率「rate」を固定する(ステップS1)。
- (2) 生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ を同時に変数として扱い、ガウスノイズ $\sigma_n$ が最大になるように、線形計画法で最適な生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ を探索する。この探索における拘束条件は、 $\lambda(1)=1$ ,  $\rho(1)=1$ と正規化し、さらに上記(10)式を満たすことである。

## 【0074】

このように、本実施の形態では、上記(9)式と上記(10)式を満たす生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ を1回の線形計画法で求めることとしたため、上記論文(chung等)のように、生成関数 $\lambda(x)$ の導出と生成関数 $\rho(x)$ の導出を繰り返し実行し、双方の最適値を求める方法よりも、容易かつ短時間に、確定的でかつ特性が安定したアンサンブルを生成できる。

## 【0075】

つぎに、上記ステップS5で求めた $b_1, b_2, \mu_1, \mu_2$ にしたがって後述する行の分割処理(ステップS8)を実行した後の行数を $M'$ とした場合、下記(14)式に示す行数を、上記図3に示す基本行列の底辺から順に削除する(ステップS6)。その結果、行削除後の行列は、列の重みのセットが $\{d_1, d_2, \dots, d_\alpha\}$ となる。

## 【0076】

【数14】

$$\frac{M' - M}{b_1 + b_2} \dots (14)$$

## 【0077】

つぎに、上記本実施の形態のガウス近似法による最適化を用いて、さらに上記で求めた $\mu_1, \mu_2, \rho\mu_1', \rho\mu_2'$ を固定のパラメータとして、暫定的に、要求された符号化率に基づく「Irregular-LDPC符号」のアンサンブルを求める(ステップS7)。そして、列の重み $x$ の数が1よりも小さくなった特定の列に関しては、その重みを候補から削除する。図5は、 $\gamma_1 = d_1 = 12$ ,  $rate = 0.5$ の場合の、重み配分後の生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ のアンサンブル(暫定)を示す図である。

## 【0078】

つぎに、上記で求めた重み配分を満たし、かつ下記の(15)式を満たす、列の重み候補のセット $\{\gamma_1, \gamma_2, \dots, \gamma_l\}$ を選択する(ステップS7)。なお、 $\gamma_1 \leq 2^s$ である。そして、下記の(15)式を満たさない列の重みが存在する場



合には、その列の重みを候補から削除する。

【0079】

【数15】

$$\begin{bmatrix} a_{1,1}^{(d_\beta)} & a_{1,1}^{(d_\beta)} & \cdots & a_{1,1}^{(d_\beta)} \\ a_{1,1}^{(d_\beta)} & a_{1,1}^{(d_\beta)} & \cdots & a_{1,1}^{(d_\beta)} \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_l \end{bmatrix} = \begin{bmatrix} d_\beta \\ d_\beta \\ \vdots \\ d_\beta \end{bmatrix} \quad \dots (15)$$

【0080】

なお、各  $a$  は、列の重み  $d_\beta$  ( $\beta = \{1, 2, \dots, \alpha\}$ ) を構成するための  $\{\gamma_1, \gamma_2, \dots, \gamma_l\}$  に対する非負の整数となる係数を表し、 $i, j$  は正の整数であり、 $\gamma_i$  は列の重みを表し、 $\gamma_l$  は列の最大重みを表す。

【0081】

つぎに、上記本実施の形態のガウス近似法による最適化を用いて、さらに上記で求めた  $\mu_1, \mu_2, \rho\mu_1', \rho\mu_2'$  と  $\{\gamma_1, \gamma_2, \dots, \gamma_l\}$  を固定パラメータとして、要求された符号化率に基づく「Irregular-LDPC符号」のアンサンブルを求める（ステップS7）。図6は、 $\gamma_l = d_l = 12$ ,  $rate = 0.5$  の場合の、重み配分後の生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンブルを示す図である。

【0082】

つぎに、分割処理を行う前に、図6に示す生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンブルの重み配分を以下の手順で調整する（ステップS7）。なお、ここでは、図6の重み配分を調整したものではないが、説明の便宜上、図7の例を用いて上記調整手順について説明する。図7は、重み配分調整用テーブルの一具体例を示す図である。

【0083】

(1) ガウス近似法で求めた生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンブルをテーブルの2列目と3列目に設定する。

(2) 重み配分  $\lambda_x$  および  $\rho_x$  (3列目) と、ユークリッド幾何符号 EG(2, 2

5) における全行列の「1」の総数  $TP = 26688$  と、を乗算し、重み単位の重み総数を求め、さらに、当該重み単位の重み総数とその総和を4列目に設定する。

(3) 重み単位の重み総数(4列目)を対応する重み  $x$  で割り、重み単位の総列数を求め、それを5列目に設定する。

(4) 重み単位の総列数が小数点以下を含む場合、丸め処理(四捨五入, 切上げ, 切捨て等)を行い、その結果を6列目に設定する。

(5) 丸め処理後の重み単位の総列数(6列目)と対応する重み  $x$  とを乗算し、丸め処理後の重み単位の重み総数を求め、それを7列目に設定する。そして、各重み総数の総和(7列目の合計の行)が行列内の「1」の総数( $TP = 26688$ )と等しいかどうかを確認する。

(6) 行列内の「1」の総数に等しくない場合、丸め処理後の重み単位の重み総数(7列目)を整数単位で調整し、その結果を8列目に設定する。この場合、8列目の総和が、行列内の「1」の総数( $TP = 26688$ )に等しくなるように調整する。

(7) 調整後の重み単位の重み総数(8列目)を対応する重み  $x$  で割り、調整後の重み単位の総列数を求め、それを9列目に設定する。調整後の各重みの配分(11列目)は、可能な限りガウス近似法で求めた値(3列目)に近い値にする。

【0084】

なお、上記調整手順は、一般的に、下記の(16), (17), (18)式のように表現することができる。まず、重み  $\gamma_i$  に属する列の数  $n_{\gamma_i}$  と、重み  $\mu_i$  に属する行の数  $n_{\mu_i}$  と、を以下の(16)式で求める。なお、 $w_t$  は行を削除した後の行列の重みの総和を示す。

【0085】

【数16】

$$n_{\gamma_i} = \text{round}\left(w_t \times \frac{\lambda_{\gamma_i}}{\gamma_i}\right), n_{\mu_i} = \text{round}\left(w_t \times \frac{\rho_{\mu_i}}{\mu_i}\right) \dots (16)$$

【0086】

そして、以下の条件（（17），（18）式）の下で、重み $\gamma_i$ に属する最終的な列の数 $n_{\gamma_i}'$ と、重み $\mu_i$ に属する最終的な行の数 $n_{\mu_i}'$ を求める。なお、 $\lambda_{\gamma_i}'$ は行の削除後の行列にあわせて調整した重み $\gamma_i$ に属する列の比率を表し、 $\rho_{\mu_i}'$ は行の削除後の行列にあわせて調整した重み $\mu_i$ に属する行の比率を表す。

【0087】

【数17】

$$n_{\gamma_i}' = w_t \times \frac{\lambda_{\gamma_i}'}{\gamma_i}, i=1,2,\dots,l$$

$$\sum_{i=1}^l n_{\gamma_i}' \cdot \gamma_i = w_t$$

$$\arg \cdot \min \sum_{i=1}^l |\lambda_{\gamma_i} - \lambda_{\gamma_i}'| \quad \dots (17)$$

【0088】

【数18】

$$n_{\mu_i}' = w_t \times \frac{\rho_{\mu_i}'}{\mu_i}, i=1,2$$

$$\sum_{i=1}^2 n_{\mu_i}' \cdot \mu_i = w_t \quad \dots (18)$$

【0089】

図8は、上記の手順で図6に示すアンサンプルを調整した場合の、ステップS7における最終的な生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ のアンサンプルを示す図である。

【0090】

最後に、ユークリット幾何符号における1行あるいは1列の分割手順（ステップS8）について説明する。たとえば、分割手順に関して、Y. Kou等の論文では、規則的に分割する方法を提示している。図9は、上記論文における分割手

順を示す図である。まず、図 9 に示すように行列のナンバリングを行う。ここでは、列番号を左端から順に 1, 2, 3, ... とし、行番号を上から順に 1, 2, 3, ... とする。そして、たとえば、3 2 点×1 列を 8 点×4 列に分割する場合、下記 (19) 式にしたがって規則的に分割する。

【0091】

【数19】

$$S_m(n) = B_1(m + 4 * n) \quad \dots (19)$$

【0092】

なお、 $m = 1, 2, 3, 4$  とし、 $n = 0, 1, 2, 3, 4, 5, 6, 7$  とし、 $l$  は  $EG(2, 2^5)$  の列番号を表す。また、 $B_1(x)$  は  $EG(2, 2^5)$  の 1 列目の「1」の位置を表し、 $S_m(n)$  は分割後の行列の  $m$  列目の「1」の位置を表す。

【0093】

具体的にいうと、 $EG(2, 2^5)$  における 1 列中の「1」の位置を示す行番号は、

$$B_1(x) = \{1 \ 32 \ 114 \ 136 \ 149 \ 223 \ 260 \ 382 \ 402 \ 438 \ 467 \ 507 \ 574 \ 579 \ 588 \ 622 \ 634 \ 637 \ 638 \ 676 \ 717 \ 728 \ 790 \ 851 \ 861 \ 879 \ 947 \ 954 \ 971 \ 977 \ 979 \ 998\}$$

となり、その結果、分割後の行列における 1～4 列目の「1」の位置を示す行番号は、 $B_1(x)$  から「1」の番号が規則的に抽出され、

$$S_1(n) = \{1 \ 149 \ 402 \ 574 \ 634 \ 717 \ 861 \ 971\}$$

$$S_2(n) = \{32 \ 223 \ 438 \ 579 \ 637 \ 728 \ 879 \ 977\}$$

$$S_3(n) = \{114 \ 260 \ 467 \ 588 \ 638 \ 790 \ 947 \ 979\}$$

$$S_4(n) = \{136 \ 382 \ 507 \ 622 \ 676 \ 851 \ 954 \ 998\}$$

となる。すなわち、3 2 点×1 列が 8 点×4 列に分割される。

【0094】

一方、本発明におけるユークリット幾何符号の分割処理は、上記のように規則的に分割するのではなく、 $B_1(x)$  から「1」の番号をランダムに抽出する（後述するランダム分割の具体例を参照）。なお、この抽出処理は、ランダム性が

保持されるのであればどのような方法を用いてもよい。

### 【0095】

これにより、分割後の行列の $m$ 列目の「1」の位置の一例を $R_m(n)$ とした場合、 $R_m(n)$ は、たとえば、

$$R_1(n) = \{1 \ 114 \ 574 \ 637 \ 851 \ 879 \ 977 \ 979\}$$

$$R_2(n) = \{32 \ 136 \ 402 \ 467 \ 588 \ 728 \ 861 \ 971\}$$

$$R_3(n) = \{149 \ 260 \ 382 \ 438 \ 579 \ 638 \ 717 \ 998\}$$

$$R_4(n) = \{223 \ 507 \ 622 \ 634 \ 676 \ 790 \ 947 \ 954\}$$

となる。

### 【0096】

ここで、上記ランダム分割の一例、すなわち、上記「乱数系列のラテン方阵を用いた分割方法」を詳細に説明する。ここでは、ランダム分割を行う場合のランダム系列を容易かつ確定的に生成する。この方法による利点は、送信側と受信側が同じランダム系列を生成できることにある。これは、現実のシステムではきわめて重要となる。また、符号特性の条件が正確に規定できる、という利点もある。

### 【0097】

(1) 基本のランダム系列を作成する。

以下に、ランダム系列作成の一例を記述する。ここでは、説明の便宜上、ユークリット幾何符号 $EG(2, 2^5)$ を用いる。ユークリット幾何符号 $EG(2, 2^5)$ の場合、1行に存在する「1」の数は $2^5 = 32$ 個である。

### 【0098】

$P$ を $P \geq 2^S$ を満たす最小の素数とした場合、たとえば、 $2^5$ のときは $P = 37$ となる。ここで、系列長 $P - 5 = 32$ の基本のランダム系列 $C(i)$ を(20)式にしたがって作成する。

$$C(1) = 1$$

$$C(i+1) = G_0 \times C(i) \mod P \quad \dots (20)$$

ただし、 $i = 0, 1, \dots, P-2$ とし、 $G_0$ はガロア体 $GF(P)$ の原始元である。その結果、 $C(i)$ は、

$C(i) = \{1 \ 2 \ 4 \ 8 \ 16 \ 32 \ 27 \ 17 \ 34 \ 31 \ 25 \ 13 \ 26 \ 15 \ 30 \ 23$   
 $9 \ 18 \ 36 \ 35 \ 33 \ 29 \ 21 \ 5 \ 10 \ 20 \ 3 \ 6 \ 12 \ 24 \ 11 \ 22$   
 $7 \ 14 \ 28 \ 19\}$

となる。

【0099】

(2) 系列長が $2^5 = 32$ となるように、32より大きい数を削除する。

$C(i) = \{1 \ 2 \ 4 \ 8 \ 16 \ 32 \ 27 \ 17 \ 31 \ 25 \ 13 \ 26 \ 15 \ 30 \ 23 \ 9$   
 $18 \ 29 \ 21 \ 5 \ 10 \ 20 \ 3 \ 6 \ 12 \ 24 \ 11 \ 22 \ 7 \ 14 \ 28 \ 19\}$

【0100】

(3) 基本のランダム系列を一定間隔で読み出すためにスキップ間隔 $S(j)$ を以下の(21)式のように定義する。

$$S(j) = j \quad j = 1, 2, \dots, 2^s \quad (21)$$

【0101】

(4) 以下の(22)式で置換パターン $LB_j(i)$ を作成する。

$$LB_j(i) = ((S(j) \times i) \bmod P) + 1$$

$$j = 1, 2, \dots, 2^s$$

$$i = 1, 2, \dots, P-1 \quad (22)$$

なお、 $LB_j(i)$ も $2^s$ より大きい数字は削除する。図10は、基本のランダム系列 $C(i)$ と基本のランダム系列の置換パターン $LB_j(i)$ を示す図である。

。

【0102】

(5)  $q$ 列 $i$ 行で $j$ 番目のラテン方阵行列 $L_{jq}(i)$ を以下の(23)式で算出することによって、分割処理を行う。このとき、ステップS6の削除処理によって、列の重み $d_\beta$ が $d_\beta < 2^s$ の場合には、 $d_\beta$ 以上の数字を $L_{jq}(i)$ の要素から間引く。

$$L_{jq}(i) = LB_j(((q+i-2) \bmod 2^s) + 1)$$

$$j = 1, 2, \dots, 2^s$$

$$i = 1, 2, \dots, 2^s$$

$$q = 1, 2, \dots, 2^s \quad (16)$$

## 【0 1 0 3】

図 1 1 は、ラテン方陣行列  $L_{jq}(i)$  を示す図である。このラテン方陣行列  $L_{jq}(i)$  は、拡張する対象の行列の  $j \times 32 + q$  列目の分割パターンを決める。たとえば、削除により短縮された  $EG(2, 2^5)$  の 670 列目  $g_{670}(1)$  を

$$g_{670}(1) = \{28 \ 48 \ 84 \ 113 \ 153 \ 220 \ 225 \ 234 \ 268 \ 280 \ 283 \ 284 \ 322 \ 363 \ 374 \ 436 \ 497 \ 507 \ 525 \ 593 \ 600 \ 617 \ 623 \ 625 \ 644 \ 670 \ 701 \ 783 \ 805 \ 818 \ 892 \ 929\}$$

とし、これを重み 6 の 5 列と重み 2 の 1 列に分割する。対応するラテン方陣  $L_{jq}(i)$  は  $20 * 32 + 30 = 670$  であるため、

$$L_{21,30}(i) = \{13 \ 19 \ 9 \ 10 \ 16 \ 24 \ 25 \ 28 \ 23 \ 5 \ 8 \ 12 \ 31 \ 14 \ 30 \ 21 \ 4 \ 6 \ 17 \ 715 \ 29 \ 2 \ 3 \ 27 \ 22 \ 26 \ 18 \ 1 \ 20 \ 32 \ 11\}$$

となる。結果として、分割パターンは以下のようになる。

$$\begin{aligned} g_{670,1}(1) &= g_{670}(L_{21,30}(1)) \\ &= \{322 \ 525 \ 268 \ 280 \ 436 \ 625\} \quad i = 1, 2, \dots, 6 \end{aligned}$$

$$\begin{aligned} g_{670,2}(1) &= g_{670}(L_{21,30}(1)) \\ &= \{644 \ 783 \ 623 \ 153 \ 234 \ 284\} \quad i = 7, 8, \dots, 12 \end{aligned}$$

$$\begin{aligned} g_{670,3}(1) &= g_{670}(L_{21,30}(1)) \\ &= \{892 \ 363 \ 818 \ 600 \ 113 \ 220\} \quad i = 13, 14, \dots, 16 \end{aligned}$$

$$\begin{aligned} g_{670,4}(1) &= g_{670}(L_{21,30}(1)) \\ &= \{497 \ 225 \ 374 \ 805 \ 48 \ 84\} \quad i = 17, 18, \dots, 24 \end{aligned}$$

$$\begin{aligned} g_{670,5}(1) &= g_{670}(L_{21,30}(1)) \\ &= \{701 \ 617 \ 670 \ 507 \ 28 \ 593\} \quad i = 25, 26, \dots, 30 \end{aligned}$$

$$\begin{aligned} g_{670,6}(1) &= g_{670}(L_{21,30}(1)) \\ &= \{929 \ 283\} \quad i = 31, 32 \end{aligned}$$

## 【0 1 0 4】

以下では、上記で説明した LDPC 符号の特性を比較する。図 1 2 は、 $E_b/N_0$  (情報 1 ビットあたりの信号電力対ノイズ電力比) と誤り率特性 (BER) との関係を示す図である。なお、復号法は「Sum-Product アルゴリズム」である。この特性は、図 8 に示すアンサンプルを使用したものであり、Y. Kou 等の論文のように規則的に分割した場合と、乱数系列のラテン方陣による

分割処理を実行した場合と、の特性比較を示す。

#### 【0105】

図12からわかるように、本実施の形態の分割方法を用いた場合には、シャノン限界に1.2dB程度まで近づいているのがわかる。このように、Y. Kou等の論文のような規則的な分割では、「Irregular-LDPC符号」であっても大幅な改善は見込めないが、本実施の形態のランダム分割を実施すると、ループの発生する確率が大幅に減るので性能が画期的に改善される。

#### 【0106】

このように、本実施の形態においては、まず、符号長と符号化率を決定し、つぎに、ベースとなるユークリット幾何符号を選択し、つぎに、選択したユークリット幾何符号を特定の関係式に基づいて並べ替えて基本行列を作成し、つぎに、条件「 $2 < \gamma_1 \leq 2^S$ 」を満たす列の重みの最大値を選択し、つぎに、行の重みの数を連続する2種類に限定した状態で、暫定的に、ガウス近似法によりIrregular-LDPC符号のアンサンブルを探索し、その後、最適な行の重みのセットを決定し、つぎに、分割後の行数を考慮して基本行列を底辺から順に削除し、つぎに、行の重みのセットを固定パラメータとして、暫定的に、ガウス近似法によりIrregular-LDPC符号のアンサンブルを探索し、その後、最適な列の重みのセットを決定し、つぎに、行の重みのセットおよび列の重みのセットを固定パラメータとして、ガウス近似法によりIrregular-LDPC符号の最適なアンサンブルを探索し、最後に、このアンサンブルに基づいて行削除後の基本行列の重みを所定の手順でランダムに分割する。これにより、確定的で特性が安定し、かつ任意のアンサンブル、任意の符号長、任意の符号化率に対応した「Irregular-LDPC符号」用の検査行列を、短時間で容易に生成できる。

#### 【0107】

なお、本実施の形態においては、「Irregular-LDPC符号」用の検査行列を生成するために、ユークリット幾何符号を用いたが、これに限らず、たとえば、Cayleyグラフを用いて、確定的な「Irregular-LDPC符号」を構成することとしてもよい。この場合、先に説明した、「ベースと



なるユークリット幾何符号を選択する処理（ステップS2）」、「選択したユークリット幾何符号を（12）式に基づいて並べ替えて基本行列を作成する処理（ステップS3）」、に置き換えて、「ベースとなるCayleyグラフの行および列の重みを決定して基本行列を作成する処理」、を実行する。これ以外の処理については、上記図1の処理と同様である。

#### 【0108】

したがって、Cayleyグラフを用いた場合であっても、上記と同様の手順を実施することによって、同様の効果を得ることができる。なお、Cayleyグラフには「J. Rosenthal, P. O. Vontobel, "Construction of LDPC codes using Ramanujan graphs and ideas from Margulis," in Proc. of the 38-th Allerton Conference on Communication, Control, Computing, 2000, pp.248-257」に示されているように、Margulisの構成法やRamanujanグラフの構成法がある。

#### 【0109】

##### 【発明の効果】

以上、説明したとおり、本発明によれば、まず、符号長と符号化率を決定し、つぎに、ベースとなるユークリット幾何符号を選択し、つぎに、選択したユークリット幾何符号を特定の関係式に基づいて並べ替えて基本行列を作成し、つぎに、列の重みの最大値を選択し、つぎに、行の重みの数を連続する2種類に限定した状態で、暫定的に、ガウス近似法によりIrregular-LDPC符号のアンサンプルを探索し、その後、最適な行の重みのセットを決定し、つぎに、分割後の行数を考慮して基本行列を底辺から順に削除し、つぎに、行の重みのセットを固定パラメータとして、暫定的に、ガウス近似法によりIrregular-LDPC符号のアンサンプルを探索し、その後、最適な列の重みのセットを決定し、つぎに、行の重みのセットおよび列の重みのセットを固定パラメータとして、ガウス近似法によりIrregular-LDPC符号の最適なアンサンプルを探索し、最後に、このアンサンプルに基づいて行削除後の基本行列の重みを所定の手順でランダムに分割する。これにより、確定的で特性が安定し、かつ任意のアンサンプル、任意の符号長、任意の符号化率に対応した「Irregular

「Irregular-LDPC符号」用の検査行列を、短時間で容易に生成できる、という効果を奏する。

#### 【0110】

つぎの発明によれば、行列内の重みが列中の上部に配置されるように並べ替えることとした。これにより、行の削除処理を行う場合に、重みの大きい列を残すことができ、かつ列内の重みのバリエーションを少なくすることができる、という効果を奏する。

#### 【0111】

つぎの発明によれば、まず、符号長と符号化率を決定し、つぎに、ベースとなるCayleyグラフの行および列の重みを決定して基本行列を作成し、つぎに、列の重みの最大値を選択し、つぎに、行の重みの数を連続する2種類に限定した状態で、暫定的に、ガウス近似法によりIrregular-LDPC符号のアンサンブルを探索し、その後、最適な行の重みのセットを決定し、つぎに、分割後の行数を考慮して基本行列を底辺から順に削除し、つぎに、行の重みのセットを固定パラメータとして、暫定的に、ガウス近似法によりIrregular-LDPC符号のアンサンブルを探索し、その後、最適な列の重みのセットを決定し、つぎに、行の重みのセットおよび列の重みのセットを固定パラメータとして、ガウス近似法によりIrregular-LDPC符号の最適なアンサンブルを探索し、最後に、このアンサンブルに基づいて行削除後の基本行列の重みを所定の手順でランダムに分割する。これにより、確定的で特性が安定し、かつ任意のアンサンブル、任意の符号長、任意の符号化率に対応した「Irregular-LDPC符号」用の検査行列を、短時間で容易に生成できる、という効果を奏する。

#### 【0112】

つぎの発明によれば、生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ を1回の線形計画法で求めることとしたため、上記論文のように、生成関数 $\lambda(x)$ の導出と生成関数 $\rho(x)$ の導出を繰り返し実行し、双方の最適値を求める方法よりも、容易かつ短時間に、確定的でかつ特性が安定したLDPC符号用の検査行列を生成することができる、という効果を奏する。

## 【0113】

つぎの発明によれば、重み配分を、重み単位の重み総数が整数で、かつ重み単位の重み総数の総和とユークリット幾何符号の「1」の総数とが等しくなるように調整する。これにより、高精度な分割処理を実現できる、という効果を奏する。

## 【0114】

つぎの発明によれば、ランダム系列のラテン方陣を作成することによって、符号特性の条件を正確に規定できる、という効果を奏する。

## 【0115】

つぎの発明によれば、まず、符号長と符号化率を決定し、つぎに、ベースとなるユークリット幾何符号を選択し、つぎに、選択したユークリット幾何符号を特定の関係式に基づいて並べ替えて基本行列を作成し、つぎに、列の重みの最大値を選択し、つぎに、行の重みの数を連続する2種類に限定した状態で、暫定的に、ガウス近似法により Irregular-LDPC 符号のアンサンブルを探索し、その後、最適な行の重みのセットを決定し、つぎに、分割後の行数を考慮して基本行列を底辺から順に削除し、つぎに、行の重みのセットを固定パラメータとして、暫定的に、ガウス近似法により Irregular-LDPC 符号のアンサンブルを探索し、その後、最適な列の重みのセットを決定し、つぎに、行の重みのセットおよび列の重みのセットを固定パラメータとして、ガウス近似法により Irregular-LDPC 符号の最適なアンサンブルを探索し、最後に、このアンサンブルに基づいて行削除後の基本行列の重みを所定の手順でランダムに分割する構成とした。これにより、確定的で特性が安定し、かつ任意のアンサンブル、任意の符号長、任意の符号化率に対応した「Irregular-LDPC 符号」用の検査行列を、短時間で容易に生成できる、という効果を奏する。

## 【0116】

つぎの発明によれば、まず、符号長と符号化率を決定し、つぎに、ベースとなる Cayley グラフの行および列の重みを決定して基本行列を作成し、つぎに、列の重みの最大値を選択し、つぎに、行の重みの数を連続する2種類に限定し

た状態で、暫定的に、ガウス近似法により Irregular-LDPC 符号のアンサンプルを探索し、その後、最適な行の重みのセットを決定し、つぎに、分割後の行数を考慮して基本行列を底辺から順に削除し、つぎに、行の重みのセットを固定パラメータとして、暫定的に、ガウス近似法により Irregular-LDPC 符号のアンサンプルを探索し、その後、最適な列の重みのセットを決定し、つぎに、行の重みのセットおよび列の重みのセットを固定パラメータとして、ガウス近似法により Irregular-LDPC 符号の最適なアンサンプルを探索し、最後に、このアンサンプルに基づいて行削除後の基本行列の重みを所定の手順でランダムに分割する構成とした。これにより、確定的で特性が安定し、かつ任意のアンサンプル、任意の符号長、任意の符号化率に対応した「Irregular-LDPC 符号」用の検査行列を、短時間で容易に生成できる、という効果を奏する。

【図面の簡単な説明】

【図 1】 本発明にかかる LDPC 符号用検査行列生成方法を示すフローチャートである。

【図 2】 選択されたユークリット幾何符号 EG (2, 2<sup>2</sup>) のマトリクスを示す図である。

【図 3】 並べ替え後のマトリクス (基本行列) を示す図である。

【図 4】 重み配分後の生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンプル (暫定) を示す図である。

【図 5】 重み配分後の生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンプル (暫定) を示す図である。

【図 6】 重み配分後の生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンプルを示す図である。

【図 7】 重み配分調整用テーブルの一具体例を示す図である。

【図 8】 図 6 に示すアンサンプルを調整後の、最終的な生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンプルを示す図である。

【図 9】 従来の論文における分割手順を示す図である。

【図 10】 基本のランダム系列 C (i) と基本のランダム系列の置換パタ

ーン  $LB_j(i)$  を示す図である。

【図 11】 ラテン方陣行列  $L_{jq}(i)$  を示す図である。

【図 12】  $E_b/N_o$  (情報 1 ビットあたりの信号電力対ノイズ電力比) と誤り率特性 (BER) との関係を示す図である。

【図 13】 LDPC 符号化／復号システムを示す図である。

【図 14】 従来の LDPC 符号用の検査行列を示す図である。

【図 15】 ユークリット幾何符号  $EG(2, 2^2)$  の構成を示す図である。

。

【図 16】 列の重みを 4 から 2 に規則的に分離した例を示す図である。

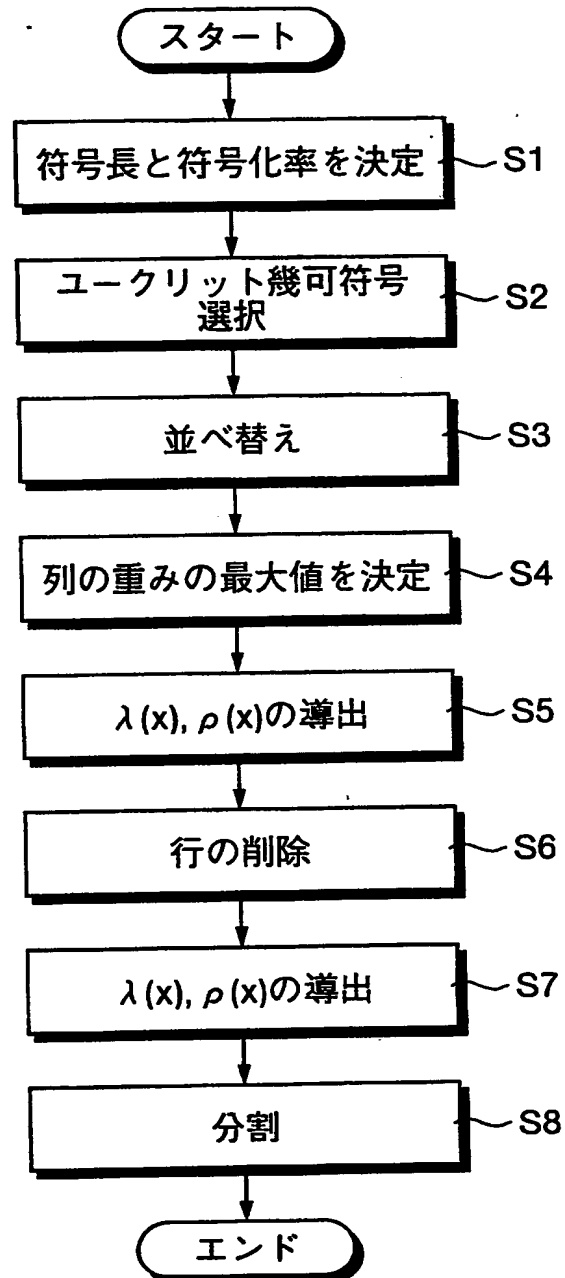
【符号の説明】

101 符号化器、102 変調器、103 通信路、104 復調器、105 復号器。

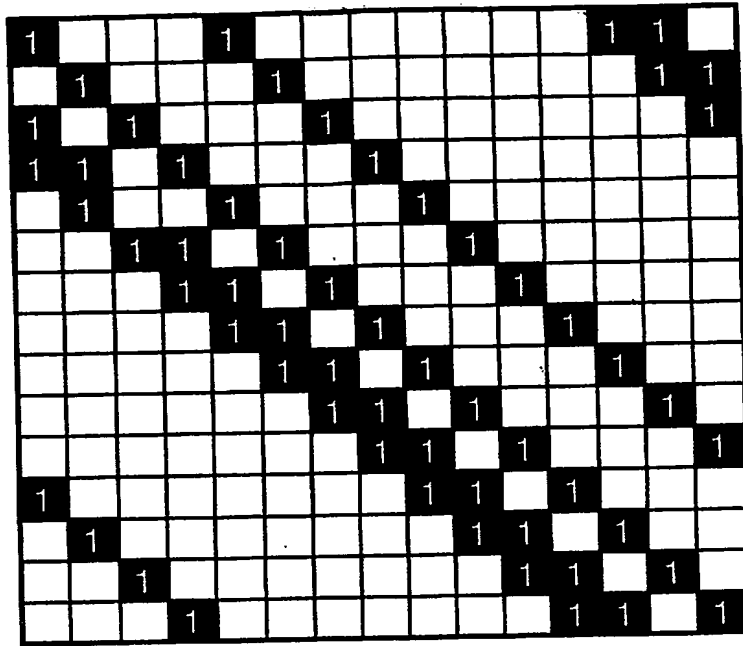
【書類名】

図面

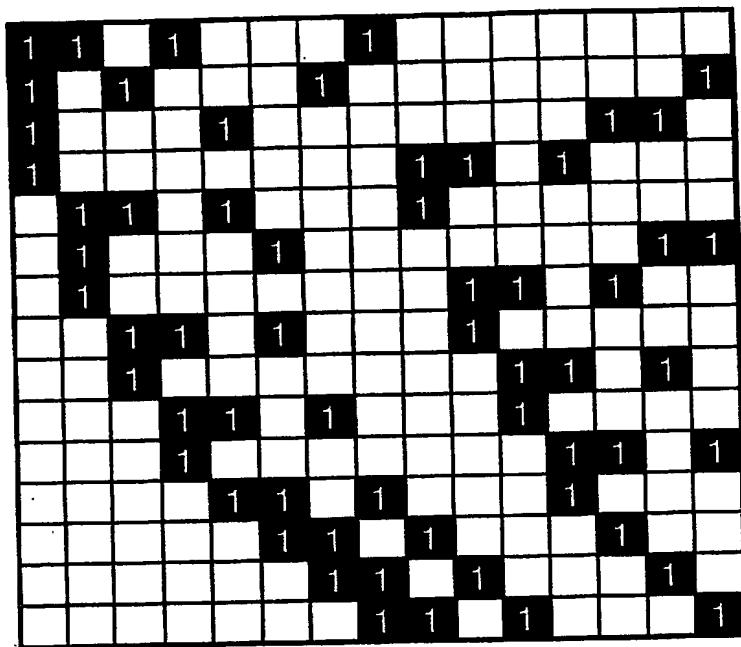
【図 1】



【図 2】



【図3】





【図 4】

rate=0.5の符号のアンサンプル

$d_i$	12	
Rate	0.5	
	$x$	$\lambda_x$
	2	0.221354
	3	0.281879
	6	0.021475
	7	0.0001
	10	8.71E-05
	12	0.475104
	$X$	$\rho_x$
	8	0.92192
	9	0.07808
$\sigma_{GA}$	0.946711	
$SNR_{norm}(GA)$	0.2886dB	

【図 5】

良好な rate=0.5 の符号のアンサンブル

$d_i$	12	
Rate	0.5	
	$x$	$\lambda_x$
	2	0.221354
	3	0.281879
	6	0.021475
	7	0.0001
	12	0.475104
	$X$	$\rho_x$
	8	0.92192
	9	0.07808
$\sigma_{GA}$	0.946736	
$SNR_{norm}(GA)$	0.2884dB	

【図 6】

良好なrate=0.5の符号のアンサンブル

$d_l$	12	
Rate	0.5	
	$x$	$\lambda_x$
	2	0.224033
	3	0.282078
	6	0.033481
	7	0.000166
	12	0.460242
	$X$	$\rho_x$
	8	1
$\sigma_{GA}$	0.947054	
$SNR_{norm}(GA)$	0.2855dB	

【図7】

1	2	3	4	5	6	7	8	9	10	11
	X	$\Lambda_x$							X	$\lambda_x$
バリアブル ノード (列)	2	0.224033399	5376.802	2688.401	2688	5376	5372	2686	2	0.223833333
	3	0.282077752	6769.866	2256.622	2257	6771	6777	2259	3	0.282375
	6	0.033481078	803.5459	133.9243	134	804	804	134	6	0.0335
	7	0.000166151	3.98762	0.56966	1	7	7	1	7	0.000291667
	12	0.46024162	11045.8	920.4832	920	11040	11040	920	12	0.46
合計		0.582003275	24000	6000	6000	23998	24000	6000		1
	X	$\rho_x$							X	$\rho_x$
チェック ノード (行)	8	1	24000	3000	3000	24000	24000	3000	8	1
		1	24000	3000	3000	24000	24000	3000		1
rate		0.500000001								0.5

・行列内の1の総数 $TP=(1023-273) \times 32=24000$

【図 8】

削除により短縮されたEG(2,2<sup>5</sup>)の拡張による  
rate=0.5のirregular LDPC符号

$d_i$	12		
符号長	6000		
	$x$	$\lambda_x$	No.
	2	0.223833	2686
	3	0.282375	2259
	6	0.0335	134
	7	0.000292	1
	12	0.46	920
	$X$	$\rho_x$	No.
	8	1	3000
$\sigma_{GA}$	0.946999		
$SNR_{norm}(GA)$	0.2860dB		

【図9】

	1	2	3	4	5	6	7	8	9	10	11	12	13
1													
2													
3													
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													

【図 10】

2

GO

J

LB(1)

LB(2)

LB(3)

LB(4)

LB(5)

LB(6)

LB(7)

LB(8)

LB(9)

LB(10)

LB(11)

LB(12)

LB(13)

LB(14)

LB(15)

LB(16)

LB(17)

LB(18)

LB(19)

LB(20)

LB(21)

LB(22)

LB(23)

LB(24)

LB(25)

LB(26)

LB(27)

LB(28)

LB(29)

LB(30)

LB(31)

LB(32)

C(1)

C(2)

C(3)

C(4)

C(5)

C(6)

C(7)

C(8)

C(9)

C(10)

C(11)

C(12)

C(13)

C(14)

C(15)

C(16)

C(17)

C(18)

C(19)

C(20)

C(21)

C(22)

C(23)

C(24)

C(25)

C(26)

C(27)

C(28)

C(29)

C(30)

C(31)

C(32)

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

Basic Random sequences

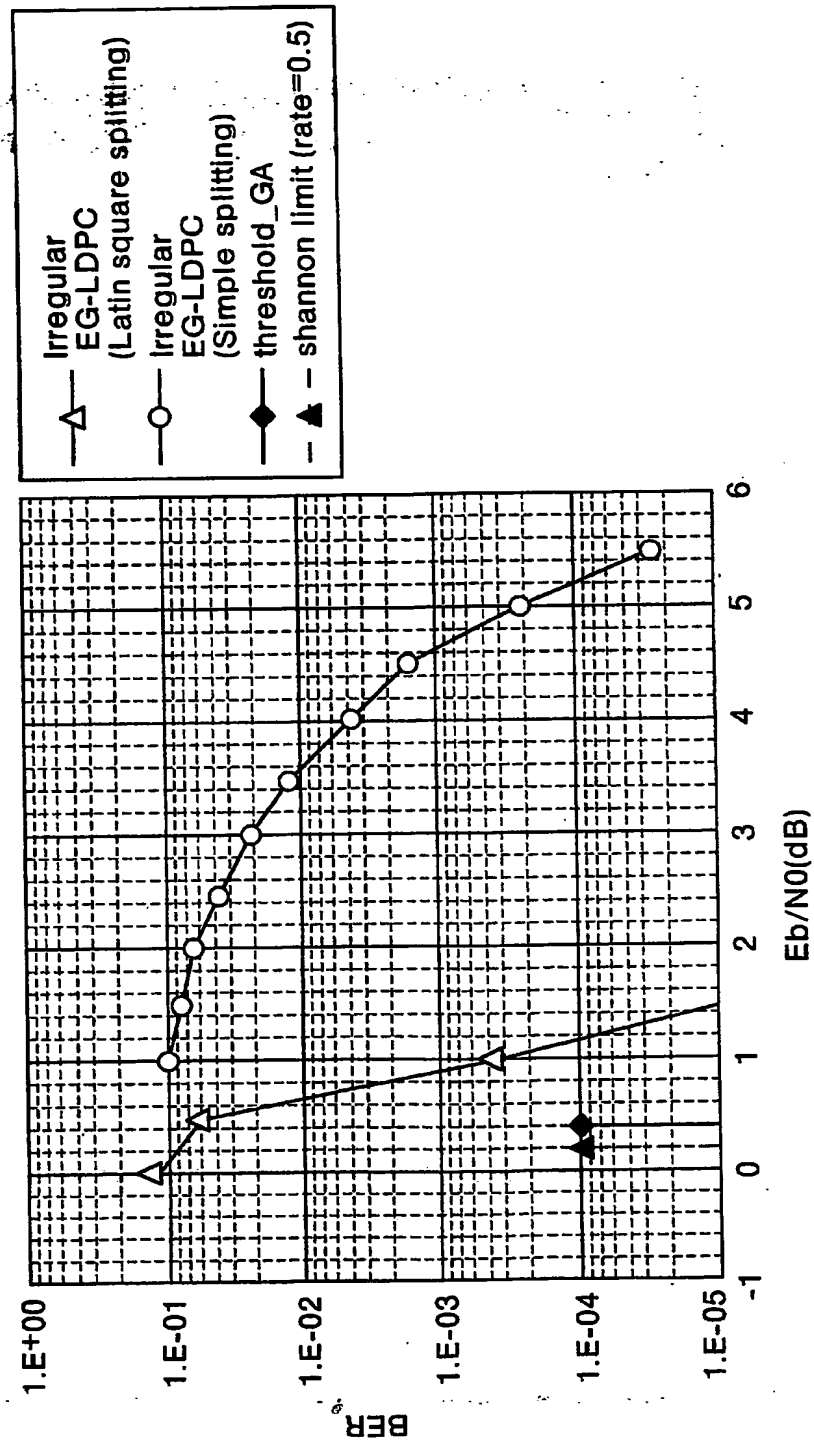
permutation pattern of basic random sequences

【図11】

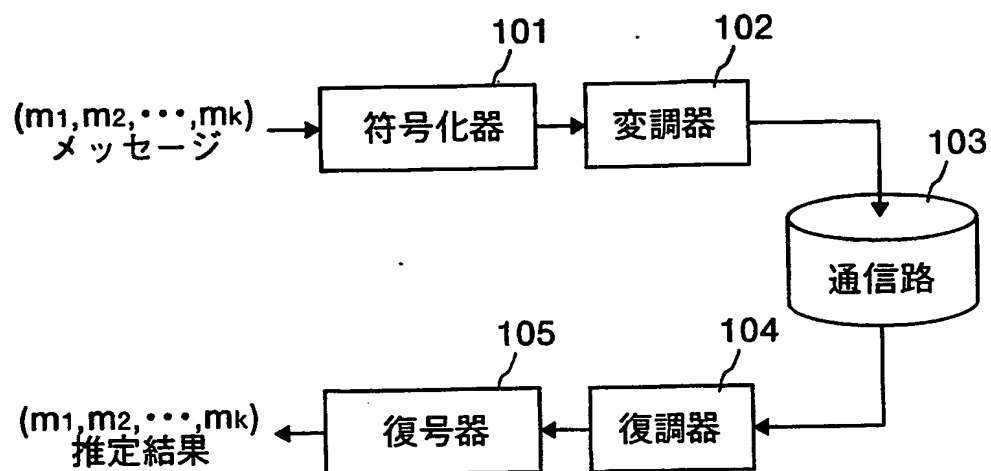
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
$q$	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9
$L_0(1)$	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10
$L_0(2)$	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16
$L_0(3)$	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24
$L_0(4)$	28	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25
$L_0(5)$	23	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28
$L_0(6)$	5	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23
$L_0(7)$	8	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5
$L_0(8)$	12	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8
$L_0(9)$	31	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12
$L_0(10)$	14	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31
$L_0(11)$	30	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14
$L_0(12)$	21	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30
$L_0(13)$	4	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21
$L_0(14)$	6	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4
$L_0(15)$	17	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6
$L_0(16)$	7	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17
$L_0(17)$	15	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7
$L_0(18)$	29	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15
$L_0(19)$	2	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28
$L_0(20)$	3	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2
$L_0(21)$	27	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3
$L_0(22)$	22	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27
$L_0(23)$	26	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22
$L_0(24)$	18	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22	26
$L_0(25)$	1	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22	26	18
$L_0(26)$	20	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22	26	18	1
$L_0(27)$	32	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22	26	18	1	20
$L_0(28)$	11	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22	26	18	1	20	32
$L_0(29)$	13	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22	26	18	1	20	32	11
$L_0(30)$	19	9	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22	26	18	1	20	32	11	13
$L_0(31)$	8	10	16	24	25	28	23	5	8	12	31	14	30	21	4	6	17	7	15	28	2	3	27	22	26	18	1	20	32	11	13	19
$L_0(32)$																																



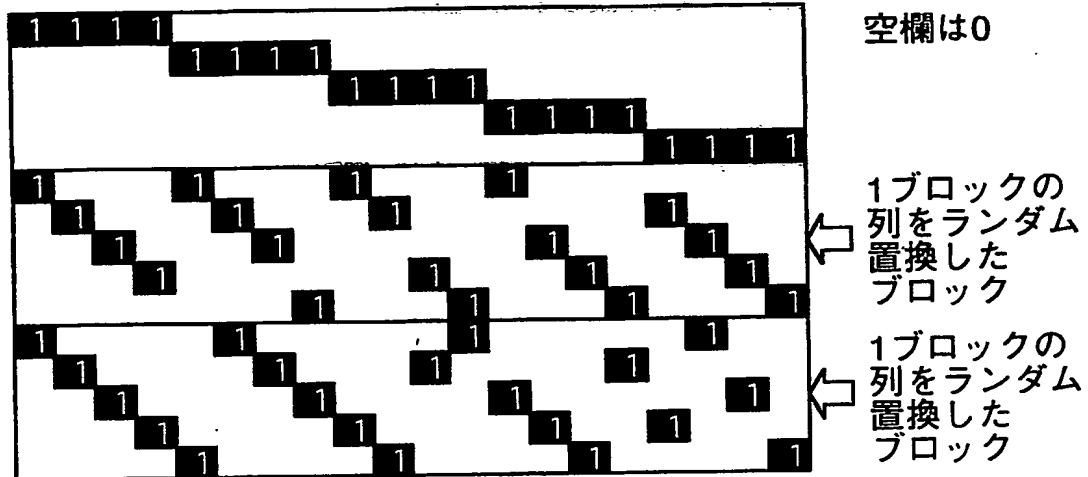
【図 12】



【図13】



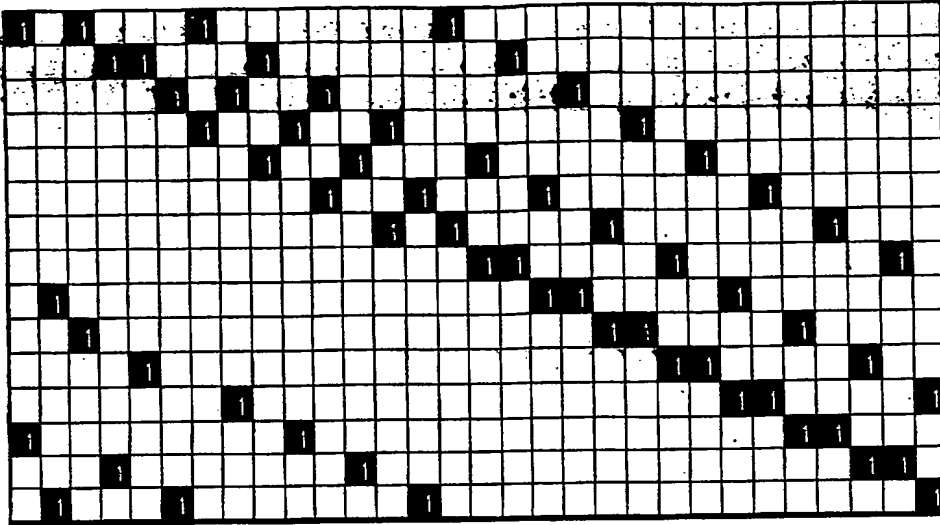
【図 1 4】



【図 1 5】

	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>4</sub>	C <sub>5</sub>	C <sub>6</sub>	C <sub>7</sub>	C <sub>8</sub>	C <sub>9</sub>	C <sub>10</sub>	C <sub>11</sub>	C <sub>12</sub>	C <sub>13</sub>	C <sub>14</sub>	C <sub>15</sub>
R <sub>1</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>2</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>3</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>4</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>5</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>6</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>7</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>8</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>9</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>10</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>11</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>12</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>13</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>14</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R <sub>15</sub>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

【図 16】



【書類名】 要約書

【要約】

【課題】 確定的で特性が安定したLDPC符号用検査行列を探索可能なLDPC符号用検査行列生成方法を得ること。

【解決手段】 本発明のLDPC符号用検査行列生成方法は、符号長、符号化率、ユークリット幾何符号を決定し、当該符号を並べ替えて基本行列を作成し、列の最大重みを選択するステップと、行の重みを連続する2種類に限定した状態で行と列の重みのアンサンブルを探索し、最適な行の重みのセットを決定するステップと、基本行列を底辺から順に削除するステップと、行の重みのセットを固定パラメータとして行と列の重みのアンサンブルを探索し、最適な列の重みのセットを決定するステップと、各重みのセットを固定パラメータとして行と列の重みの最適なアンサンブルを探索するステップと、最適なアンサンブルに基づいて行削除後の基本行列の重みをランダムに分割するステップと、を含むことを特徴とする。

【選択図】 図1

特願2002-193767

出願人履歴情報

識別番号

[000006013]

1. 変更年月日

1990年 8月24日

[変更理由]

新規登録

住所

東京都千代田区丸の内2丁目2番3号

氏名

三菱電機株式会社